

contents

0.1	Brief introduction	1
0.2	Update Diary	1
0.3	Structure of the code	2
0.4	Misc	2

0.1 Brief introduction

This code is used for designing 3-D coils in stellarators, knotrans and tokamaks. FOCUS (Finding/Flexible Optimized Coils Using Space curves) gets rid of winding surface by representing coils using space curves (either Fundamental theorem of space curves or Fourier series or other representations.). And for the first time, the derivatives (both the first and the second ones) are analytically calculated.

Parts of the code were first written by [Dr. Stuart R. Hudson](#) in April 2016. Then Caoxiang Zhu (CZHU) took over the whole project and it's currently under developping. If you have any questions, please send a email to czhu@pppl.gov (or zcxiang@mail.ustc.edu.cn).

0.2 Update Diary

2015/10/30: Dr. Stuart Hudson wrote the code **OPTIM** using [NAG:E04JYF](#) to find the optimal Fourier series for coils on a given winding surface.

2016/04/xx: New code **KNOTOPT** that represents coils using 3D Fourier series was written.

2016/04/21: CZHU began to join the project and mainly took over the project.

2016/xx/xx: A lot of new stuffs were added in but not well documented.

2016/11/01: The code was renmaed to FOCUS and a poster was presented by CZHU at the APS-DPP meeting in San Jose, CA.

2017/02/15: A re-writing for debugging and better structure began by CZHU.

2017/04/04: The code repository was tranported to Princeton University @ GitHub

2017/05/15: Nonlinear Conjugate Gradient method was implemented.

2017/05/20: Truncated Newton Method with Preconditioning CG method was implemented.

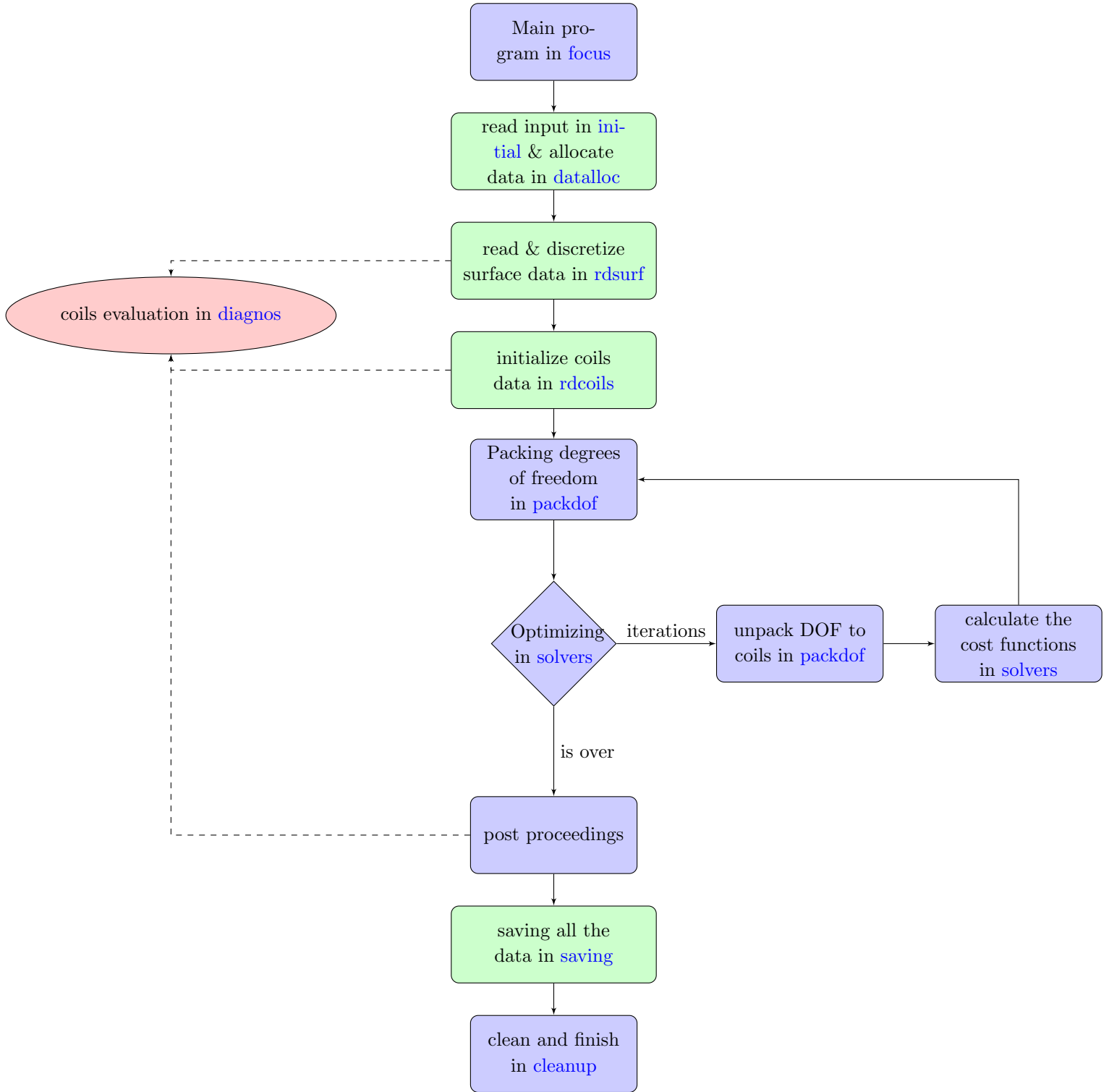
2017/06/04: The first paper introducing FOCUS was submitted to Nuclear Fusion.

2017/06/07: Hybrid Newton method was implemented.

2017/06/23: NAG and OCULUS dependance have been removed in the new code.

2017/07/18: Enable field periodicity and add coil diagnostic part.

0.3 Structure of the code



0.4 Misc

- Details about the macros used are listed in *Macros*;
- The unit used in FOCUS is the International System of Units (SI);
- All the shared variables are defined in [globals](#);
- Variables starting with **case_**xxx are used for controlling the flow direction;
- Variables starting with **Is**xxx are used for logical judgement.